VSURF : un package R pour la sélection de variables à l'aide de forêts aléatoires

Robin Genuer [1] & Jean-Michel Poggi [2] & Christine Tuleau-Malot [3]

[1] *Université de Bordeaux, ISPED, INSERM U-897, INRIA équipe SISTM*
Robin.Genuer@isped.u-bordeaux2.fr
[2] *Université d'Orsay, Lab. de Mathematiques, bât. 425*
Jean-Michel.Poggi@math.u-psud.fr
[3] *Université de Nice Sophia Antipolis, CNRS, LJAD, UMR 7351, France*
Malot@unice.fr

Résumé. Dans cette présentation, nous décrivons VSURF, un package R. Basé sur les forêts aléatoires, il fournit deux sous-ensembles de variables associés à deux objectifs de sélection de variables pour des problèmes de régression et de classification. Le premier est un sous-ensemble de variables importantes pour l'interprétation. Le second est un sous-ensemble parcimonieux à l'aide duquel on peut faire de bonnes prédictions. La stratégie générale est basée sur un classement préliminaire des variables donné par l'indice d'importance des forêts aléatoires, puis utilise un algorithme d'introductions ascendantes de variables pas à pas. Les deux sous-ensembles peuvent être obtenus automatiquement en gardant le comportement par défaut du package, mais peuvent également être réglés en jouant sur plusieurs paramètres. Nous illustrons la méthode sur plusieurs jeux de données réelles.

Mots-clés. forêts aléatoires, sélection de variables, package R

Abstract. This paper describes the R package VSURF. Based on random forests, it delivers two subsets of variables according to two types of variable selection for classification or regression problems. The first is a subset of important variables which are relevant for interpretation, while the second one is a subset corresponding to a parsimonious prediction model. The strategy is based on a preliminary ranking of the explanatory variables using the random forests permutation-based score of importance and proceeds using a stepwise ascending variable introduction strategy. The two proposals can be obtained automatically using data-driven default values, good enough to provide interesting results, but can also be fine-tuned by the user. The algorithm is illustrated on a simulated example and its applications to real datasets are presented.

Keywords. random forests, variable selection, R package

1 Introduction

Variable selection is a crucial issue in many applied classification and regression problems (see Hastie et al., 2001). It is of interest for statistical analysis as well as for modelization or prediction purposes to remove irrelevant variables, to select all important ones or to determine a sufficient subset for prediction. These main different objectives on a statistical learning perspective involve variable selection to simplify statistical problems, to help diagnosis and interpretation, and to speed up data processing.

The authors have proposed a variable selection method based on random forests (see Genuer et al., 2010), and the aim of this paper is to describe the recently available associated R (R Core Team, 2013) package called **VSURF** and to illustrate its use on real datasets.

Introduced by Breiman (2001), random forests (abbreviated RF in the sequel) is an attractive nonparametric statistical method to deal with such problems, since it requires only mild conditions on the model supposed to have generated the observed data. Indeed, since it is based on decision trees and it uses aggregation ideas, RF allow to consider in an elegant and versatile framework different models and problems, namely regressions, two-class or multiclass classifications.

Considering a learning set $L = \{(X_1, Y_1), \ldots, (X_n, Y_n)\}$ supposed to be independent observations of the random vector (X, Y), we distinguish as usual the predictors (or explanatory variables), collected in the vector $X = (X^1, ..., X^p)$ where $X \in \mathbb{R}^p$, from the explained variable $Y \in \mathcal{Y}$ where Y is either a class label for classification problems or a numerical response for regression ones. Let us recall that a classifier t is a mapping $t : \mathbb{R}^p \to \mathcal{Y}$ while the regression function appears naturally to be the function s when we suppose that $Y = s(X) + \varepsilon$ with $E[\varepsilon|X] = 0$. Then random forests provide estimators of either the Bayes classifier, which minimizes $P(Y \neq t(X))$ the classification error, or the regression function.

The CART (Classification And Regression Trees) method defined by Breiman et al. (1984) is a well-known way to design optimal single trees. It proceeds by performing first a growing step and then a pruning one. The principle of random forests is to aggregate many binary decision trees coming from two random perturbation mechanisms: the use of bootstrap samples of L instead of L and the random choice at each node of a subset of explanatory variables instead of all of them. There are two main differences with respect to CART trees: in the growing step, at each node, a fixed number of input variables are randomly chosen and the best split is calculated only among them and no pruning is performed so all the trees of the forest are maximal trees. RF algorithm is a very popular machine learning algorithm and appears to be powerful in a lot of different applications, see for example Verikas et al. (2011) for a recent survey.

A lot of variable selection procedures are based on the cooperation of variable importance for ranking and model estimation to generate, evaluate and compare a family of models, in particular in the family of "wrapper" methods (Kohavi and John, 1997) which

include the prediction performance in the score calculation, a lot of methods can be cited. We choose to highlight one of them since it is widely used and our procedure is a generalization of it. Díaz-Uriarte and Alvarez De Andres (2006) propose a strategy based on recursive elimination of variables, with a crucial parameter (the proportion of variables to eliminate at each step). Let us emphasize that we propose an heuristic strategy which does not depend on specific model hypotheses but based on data-driven thresholds to take decisions.

This topic of variable selection still continue to be interesting, indeed recently Hapfelmeier and Ulm (2012) propose a new variable selection approach using random forests and, more generally, a survey paper Cadenas et al. (2013) describe and compare such different approaches.

Some packages are available to cope with variable selection problems. Let us cite, for classification problems the R package **Boruta**, described in Kursa and Rudnicki (2010), finding all relevant variables using a random forest classification algorithm which removes iteratively the variables using a statistical test. The R package **ofw** (see Lê Cao and Chabrier, 2008), also dedicated to the context of classification, selects relevant variables with the application of supervised multiclass classifiers such as classification and regression trees or support vector machines.

2 The strategy

In Genuer et al. (2010) we have distinguished two variable selection objectives: interpretation and prediction. The first is to find important variables highly related to the response variable in order to select all the important variables, even with high redundancy. The second is to find a small number of variables sufficient to a good parsimonious prediction of the response variable.

We have proposed the following two-steps procedure, the first one is the same for the two situation while the second one depends on the objective:

- Step 1. Preliminary elimination and ranking:
  - Compute the RF scores of importance, order the variables in decreasing order of importance;
  - Cancel the variables of small importance (let m denotes the number of remaining variables).

- Step 2. Variable selection:
  - For *interpretation*: construct the nested collection of RF models involving the k first variables, for $k = 1$ to m and select the variables involved in the model leading to the smallest OOB error;
  - For *prediction*: starting from the ordered variables retained for interpretation, construct an ascending sequence of RF models, by invoking and testing the variables stepwise. The variables of the last model are selected.

3 Two illustrative examples

In this section we experiment the proposed procedure on two real-life examples: one high dimensional dataset (associated with a classification problem) and a standard one (associated with a regression problem), to illustrate the versatility of the procedure.

3.1 Ozone data

The ozone dataset consists of $n = 366$ observations of 12 independent variables and 1 dependent variable. These variables are numbered as in the R package **mlbench**: 1-Month, 2-Day of month, 3-Day of week, 5-Pressure height, 6-Wind speed, 7-Humidity, 8-Temperature (Sandburg), 9-Temperature (El Monte), 10-Inversion base height, 11-Pressure gradient, 12-Inversion base temperature, 13-Visibility, for independent variables and 4-Daily maximum one-hour-average ozone, for the dependent variable.

The interest of this dataset is that it has been extensively studied and that even if it is a real one, it is possible to a priori know the expected variables. Moreover, this dataset, which is not a high dimensional one, includes some missing data and allows us to give an example of how to handle such data using **VSURF**.

To begin, we load the data:

```
library(VSURF)
library(mlbench)
data(Ozone)
```

Then, we apply the complete procedure via `VSURF`.

```
vozone <- VSURF(formula = V4 ~ ., data = Ozone, na.action = na.omit)
```

```
summary(vozone)

##
##  VSURF computation time: 1.7 mins
##
##  VSURF selected:
##  9 variables at thresholding step (in 55.7 secs)
##  6 variables at interpretation step (in 28.6 secs)
##  6 variables at prediction step (in 19.6 secs)
```

Let us now examine the results of the selection procedures. To reflect the renumbering of the variables in the definition of explanatory dataset, we must change the number of output variables of the procedure.

```
number <- c(1:3, 5:13)
number[vozone$varselect.thres]

## [1]  9  8 12  1 11  7  5 10 13
```

After the first elimination step, the 3 variables of very small importance (variables 6, 3 and 2) are canceled, as expected.

Then, the interpretation procedure leads to select the model with 6 variables, which contains all the most important variables: (9 8 12 1 11 7), see `number[vozone$varselect.interp]`.

Finally, the prediction set is the same as the previous one for this example, see `number[vozone$varselect.pred]`.

3.2 SRBCT data

The dataset we consider here will allow us to apply our procedure in a classification framework.

The real classification dataset considered is relative to small round blue cell tumors of childhood. This set is composed of :

- a data frame, called `gene`, of size 63×2308 which contains the 2308 gene expression;
- a response factor of length 63, called `class`, indicating the class of each sample (4 classes in total).

Those data, available in the R package **mixOmics** (Dejean et al., 2013), have been widely studied but in most of the study only 200 genes where considered and preprocessing where performed to be reduced to a regression problem (see for example Lê Cao and Chabrier, 2008).

As in Díaz-Uriarte and Alvarez De Andres (2006), we consider the 2308 genes and we deal with a classification framework.

```
library(VSURF)
library(mixOmics)
data(srbct)
```

```
vSRBCT <- VSURF(x = srbct$gene, y = srbct$class)
```

```
summary(vSRBCT)

##
##  VSURF computation time: 2.7 hours
##
```

```
##  VSURF selected:
##  651 variables at thresholding step (in 7.2 mins)
##  25 variables at interpretation step (in 2.6 hours)
##  13 variables at prediction step (in 14.6 secs)
```

On this dataset, the procedure leads to 25 and 13 selected variables after the interpretation and prediction step respectively, and the number of selected variables as the selected variables themselves are stable. We can compare these results with those obtained in Díaz-Uriarte and Alvarez De Andres (2006) where the authors select respectively 22 genes and their number of selected variables is quite stable.

Bibliographie

[1] Breiman, L. (2001). Random forests. *Machine learning*, 45(1):5–32.

[2] Breiman, L., Friedman, J., Olshen, R., and Stone, C. (1984). *Classification and Regression Trees*. Chapman & Hall, New York.

[3] Cadenas, J. M., Carmen Garrido, M., and Martínez, R. (2013). Feature subset selection filter-wrapper based on low quality data. *Expert Systems with Applications*, 40(16):6241–6252.

[4] Dejean, S., Gonzalez, I., Le Cao, KA with contributions from Monget, P., Coquery, J., Yao, F., and Liquet, B. (2013). ***mixOmics****: Omics Data Integration Project*. R package version 4.1-5.

[5] Díaz-Uriarte, R. and Alvarez De Andres, S. (2006). Gene selection and classification of microarray data using random forest. *BMC Bioinformatics*, 7(1):3.

[6] Genuer, R., Poggi, J.-M., and Tuleau-Malot, C. (2010). Variable selection using random forests. *Pattern Recognition Letters*, 31(14):2225–2236.

[7] Hapfelmeier, A. and Ulm, K. (2012). A new variable selection approach using random forests. *Computational Statistics & Data Analysis*, 60:50–69.

[8] Hastie, T., Tibshirani, R., and Friedman, J. H. (2001). *The Elements of Statistical Learning*. Springer-Verlag, New York.

[9] Kohavi, R. and John, G. H. (1997). Wrappers for feature subset selection. *Artificial Intelligence*, 97(1):273–324.

[10] Kursa, M. B. and Rudnicki, W. R. (2010). Feature selection with the **Boruta** package. *Journal of Statistical Software*, 36(11).

[11] Lê Cao, K.-A. and Chabrier, P. (2008). **ofw**: An r package to select continuous variables for multiclass classification with a stochastic wrapper method. *Journal of Statistical Software*, 28(9).

[12] R Core Team (2013). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria.

[13] Verikas, A., Gelzinis, A., and Bacauskiene, M. (2011). Mining data with random forests: A survey and results of new tests. *Pattern Recognition*, 44(2):330–349.